

Q1. What will be the output of the program? [8 Marks]

```
//RunnableExample.java
class Count implements Runnable
{
    Thread mythread ;
    Count()
    {
        mythread = new Thread(this, "my Runnable thread");
        System.out.println("my thread created");
        mythread.start();
    }
    public void run()
    {
        try
        {
            for (int i=0 ; i < 5; i++)
            {
                System.out.println("Printing the count " + i);
                Thread.sleep(1000);
            }
        }
        catch (InterruptedException e)
        {
            System.out.println("my thread interrupted");
        }
        System.out.println("mythread run is over" );
    }
}

class RunnableExample
{
    public static void main(String args[])
    {
        Count cnt = new Count();
        try
        {
            while (cnt.mythread.isAlive())
            {
                System.out.println("Main thread will be alive till the child thread is live");
                Thread.sleep(1500);
            }
        }
        catch (InterruptedException e)
        {
            System.out.println("Main thread interrupted");
        }
        System.out.println("Main thread run is over" );
    }
}
```

C:\Program Files (x86)\Xinox Software\JCreatorV4\GE20... — □ ×

```
my thread created
Main thread will be alive till the child thread is live
Printing the count 0
Printing the count 1
Main thread will be alive till the child thread is live
Printing the count 2
Main thread will be alive till the child thread is live
Printing the count 3
Printing the count 4
Main thread will be alive till the child thread is live
mythread run is over
Main thread is over
Press any key to continue..._
```


Q2. What will be the output of the program? [2 Marks]

```
1 import java.lang.*;  
2  
3 public class isAliveExample implements Runnable {  
4     public void run() {  
5  
6         Thread t = Thread.currentThread();  
7  
8         System.out.println("status = " + t.isAlive());  
9     }  
10  
11     public static void main(String args[]) throws Exception {  
12  
13         Thread t = new Thread(new isAliveExample());  
14  
15         t.start();  
16  
17         t.join();  
18  
19         System.out.println("status = " + t.isAlive());  
20     }  
21 }  
22
```



C:\Program Fi...



status = true

status = false

Press any key to continue...

